Marco Russo
marco@sqlbi.com

DAX Optimization Examples
Thanks to our Sponsors!

**Global SQLSaturday Partner**

Microsoft

**Gold Sponsor**

DELL EMC  COZYROC  PureStorage  NetLib Security

**Silver Sponsor**

SolidQ  cardinal  SQLskills  Idera  Quest

**Bronze Sponsor**

SentryOne  SQLskills

**Platinum Sponsor**

SIOS  Rubrik
We write Books

We teach Courses

We provide Consulting

We are recognized BI Experts
Agenda

- Optimize reports step-by-step
- In the meantime find time to:
  - Learn the basics of tools to measure performance
  - Understand the way DAX solves your queries
  - Learn about Formula Engine and Storage Engine
  - Measure performance and detect bottlenecks
  - Reproduce the issue and measure it
  - Solve the problems, one at a time

- No “wow effects”... we simulate real-world examples
Open a slow report for the first time...

First look
<table>
<thead>
<tr>
<th>Continent</th>
<th>CY 2007</th>
<th>CY 2008</th>
<th>CY 2009</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Europe</td>
<td>144,117</td>
<td>132,467</td>
<td>65,413</td>
<td>335,666</td>
</tr>
<tr>
<td>North America</td>
<td>276,132</td>
<td>152,669</td>
<td>223,330</td>
<td>645,543</td>
</tr>
<tr>
<td>Total</td>
<td>804,063</td>
<td>498,766</td>
<td>385,314</td>
<td>1,663,351</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Year</th>
<th>Amount USD</th>
<th>Amount EUR</th>
</tr>
</thead>
<tbody>
<tr>
<td>2008</td>
<td>$1,010,803,395.04</td>
<td>€ 739,217,874.92</td>
</tr>
<tr>
<td>2009</td>
<td>$849,671,203.42</td>
<td>€ 565,233,603.09</td>
</tr>
<tr>
<td>Total</td>
<td>$2,718,202,629.81</td>
<td>€ 1,928,482,416.38</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Education</th>
<th>Blanks</th>
<th>Clerical</th>
<th>Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>High School</td>
<td>[Bar Graph]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Partial College</td>
<td>[Bar Graph]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Partial High School</td>
<td>[Bar Graph]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Graph 1:** Open Orders by Continent and Year

**Graph 2:** Sales by Year in USD and EUR

**Graph 3:** Open Orders by Education

**Graph 4:** Open Orders Amount USD and Amount EUR by Year and Month

**Graph 5:** Occupation (% of Total Sales)
Where is my problem?

- A simple report like this, is already too complex
- Three measures:
  - Amount USD, Amount EUR, Open Orders
- Different ways of slicing and combining values
- Trial and error is not the way to go
DAX Studio

- [http://daxstudio.codeplex.com](http://daxstudio.codeplex.com)
  - Free add-in for Excel
  - Standalone executable for Tabular and Power BI
- It can query
  - Analysis Services databases
  - Power Pivot data models
  - Power BI Desktop data models
  - Both DAX and MDX queries
Main Features

- DAX Query Editor
  - Full metadata available
  - IntelliSense enabled
  - Format DAX code
    - Through www.daxformatter.com website
- Catch Query Plan and Server Timings
- Save query plan and timings for later analysis
Demo

DAX Studio
Capturing Power BI queries using DAX Studio

- Create an empty page in Power BI
- Save Power BI file selecting empty page as the current one
- Close and open Power BI file
- Connect DAX Studio to Power BI
- Activate trace for All Queries in DAX Studio
- Switch report page in Power BI
Know your data model first

- Do you know..
  - The size in memory of each column
  - The cardinality of each table
  - The cardinality of each column
    - Hint: they are not the same!

- VertiPaq Analyzer
  - Simplify extraction of this data from DMVs
Demo

VertiPaq Analyzer
Focus on DAX to search for a better way to write currency conversion

Currency Conversion
Following best practices

- Best practices are always helpful
  - there are not so many in DAX
  - it is easier to follow them
- Use variables as much as you can to fix values
- Avoid computing the same value multiple times
- Make it simpler, then simpler, then... simpler
Fighting with the cache

- There are multiple cache levels
  - they are your enemies as of now
- Power BI cache
  - Useful when authoring reports
  - Avoids executing the query at all if not needed
- DAX cache
  - Caches results of internal VertiPaq queries (more on this later)
  - Can be cleared from within DAX Studio
- Avoid the cache to measure performance
Create an empty page in Power BI

- An empty page lets you start Power BI Desktop without computing any report
- Open Power BI on the empty page
- Open DAX Studio and connect to Power BI
- Start tracing all queries
- Collect the results
- Stop the tracing
The two engines in DAX

- **QUERY (MDX/DAX)**
  - Analysis Services 2016 Tabular

- **DAX CALCULATION ENGINE**
  - Formula Engine

- **VERTIPAQ (xmSQL)**
  - Cached Data
  - PERIODIC REFRESH

- **DIRECTQUERY (SQL)**
  - Storage Engine

- **Data Source**
The two engines in DAX

- Query (MDX/DAX) to Analysis Services 2016 Tabular
- DAX Calculation Engine
  - Formula Engine
  - DirectQuery (SQL)
  - Vertipaq (xmSQL)
  - Cached Data
  - Periodic Refresh
- Data Source
### Differences between the engines

<table>
<thead>
<tr>
<th>Formula Engine</th>
<th>Storage Engine</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Computes any DAX function</td>
<td>- Computes simple expressions</td>
</tr>
<tr>
<td>- Very rich in terms of expressivity</td>
<td>- Uses model relationships</td>
</tr>
<tr>
<td>- Single threaded</td>
<td>- Multi threaded</td>
</tr>
<tr>
<td>- Slow side of DAX</td>
<td>- Fast side of DAX</td>
</tr>
</tbody>
</table>

**Performance**
What is a Data Cache?

Column Storage

<table>
<thead>
<tr>
<th>Prod</th>
<th>Amt</th>
<th>Qty</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

VertiPaq query (xmSQL)

<table>
<thead>
<tr>
<th>Prod</th>
<th>SUM(Amt)</th>
<th>SUM(Qty)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Result

<table>
<thead>
<tr>
<th>Prod</th>
<th>SUM(Amt)</th>
<th>SUM(Qty)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Storage Engine (VertiPaq)

Data Cache

<table>
<thead>
<tr>
<th>Prod</th>
<th>SUM(Amt)</th>
<th>SUM(Qty)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Formula Engine

DAX Query Plan

SE fills data cache

FE works on data caches
xmSQL

- Storage engine uses xmSQL
  - Syntax similar to SQL
  - Few options (compared to SQL)
  - Not intended for the user
- Simple mathematical operations
- Some (not all) aggregations
- Usage of model relationships
How FE uses Data Caches

- There is no indication of which data cache is used by which FE operator
- Only common sense and some geeky attitude helps
- For simple plans, it is easy
- For complex plans, it is nearly impossible
- Thus: optimize simple plans!
Next step is optimizing the open orders measure

Open Orders
October, 2017

Open Orders

Order 1
Order 2
Order 3
Order 4
Order 5

Open Orders = 3
Changing the data model

Order 1
Order 2
Order 3
Order 4
Order 5

October, 2017
Conclusions

Create a repro query
Isolate MDX or DAX query
Identify DAX expression (usually a measure)

Modify DAX expression and test different timings
DAX Studio improves productivity
Use local DAX measure overriding the one defined in the data model

At [www.sqlbi.com](http://www.sqlbi.com) you will find plenty of resources on DAX
References

- The Definitive Guide to DAX
  By Marco Russo, Alberto Ferrari – Microsoft Press
  http://www.sqlbi.com/books/the-definitive-guide-to-dax/

- Understanding DAX Query Plans
  - http://www.sqlbi.com/articles/understanding-dax-query-plans/

- Understanding Distinct Count in DAX Query Plans
Tools

- DAX Studio
  - http://daxstudio.codeplex.com

- VertiPaq Analyzer

- OLAP PivotTable Extensions
  - http://olappivottableextend.codeplex.com/
Thank you!

Check our articles, whitepapers and courses on

www.sqlbi.com